END
DATE
FILMED
7-84
DTIC

MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

Unclassified

An Algorithm for Seam Tracking Applications

P. K. Khosla, C. P. Neuman, and F. B. Prinz

CMU-RI-TR-84-6

JUN 1, 1984

A

Carnegie-Mellon University
The Robotics Institute
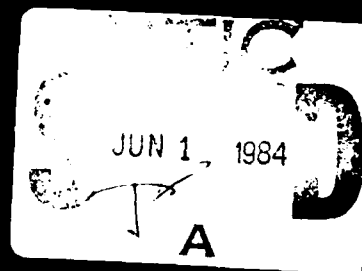
THE ROBOTICS INSTITUTE

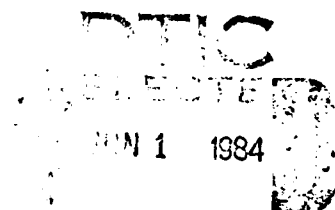# Technical Report

DTIC FILE COPY

84 05 30 100

An Algorithm for Seam Tracking Applications

P. K. Khosla, C. P. Neuman, and F. B. Prinz

CMU-RI-TR-84-6

The Robotics Institute
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

May 1984

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER CMU-RI-TR-84-6	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Algorithm for Seam Tracking Applications		5. TYPE OF REPORT & PERIOD COVERED Interim
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) P. K. Khosla, C. P. Neuman, and F. B. Prinz		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Carnegie-Mellon University The Robotics Institute Pittsburgh, PA. 15213		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE May 1984.
		13. NUMBER OF PAGES 32
14. MONITORING AGENCY NAME & ADDRESS(If different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Approved for public release; distribution unlimited

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Seam tracking is currently accomplished by special features of the robot and "a priori" knowledge of seam geometry. In this paper we demonstrate the feasibility of tracking a seam in real-time. A general-purpose seam tracking algorithm is developed for implementation on a robot with six degress-of-freedom. The algorithm is motivated by a physical interpretation of the T_6 matrix and dT_6 matrices, and the assumption that 3-D seam data are available. In the past, the dT_6 matrix and inverse Jacobian solutions have been used to compute the differential changes in the joint variables. The versatile seam tracking algorithm can be applied to a

(cont'd)

multitude of robotic seam tracking activities such as gluing, surface grinding
and flame cutting.

Table of Contents

A-1

List of Figures

List of Tables

Abstract

Seam tracking is currently accomplished by special features of the robot and *a priori* knowledge of seam geometry. In this paper we demonstrate the feasibility of tracking a seam in real-time. A general-purpose seam tracking algorithm is developed for implementation on a robot with six degrees-of-freedom. The algorithm is motivated by a physical interpretation of the T_6 and dT_6 matrices, and the assumption that 3-D seam data are available. In the past, the dT_6 matrix and inverse Jacobian solutions have been used to compute the differential changes in the joint angles. By using the inverse Jacobian, an iterative algorithm is introduced to compute both large and small changes in the joint variables. The versatile seam tracking algorithm can be applied to a multitude of robotic seam tracking activities such as gluing, surface grinding and flame cutting.

1. Introduction

Manufacturing operations such as robotic welding, gluing, sealing and surface grinding require trajectory control of the tool mounted on the end-effector of the robot. While the kinematic control algorithm developed in this paper is applicable to a multitude of manufacturing operations, robotic arc welding nomenclature is used. The tool is called the welding torch, and the tool trajectory is defined by the weld seam. The fundamental problem is to position a welding gun with the proper translational and rotational positions with respect to a curved weld seam in three dimensional (3-D) space. A seam in space can be traced by a five degree-of-freedom robot. Since real-time seam tracing requires that *both* the sensor *and* torch trace the seam, a six degree-of-freedom robot is required. Automation of the welding process thus demands a robot with *six* degrees-of-freedom and acceptable performance in terms of speed and accuracy. The control task also demands a *real-time* algorithm to guide the robot in a fixed geometrical relationship to the contour.

Automation of the welding process can be divided into two distinct components :

- Seam data acquisition and interpretation by a sensor system ; and

- Guidance and control of the robot to traverse the seam, while maintaining the proper orientation and position.

Since the seam data constitute the 3-D coordinates of the seam, a visual sensory device (such as a light stripe imaging device) can be used to obtain the seam data. Methods for obtaining 3-D data from 2-D images have been developed [Agin 82]. Having obtained and interpreted the seam data, the control system must guide the robot to traverse the seam and place the requisite amount of weld material along the path. The torch position and attitude must be controlled precisely to ensure acceptable weld quality.

Seam tracking is currently accomplished by exploiting a special feature of the robot or limiting the application to a particular type of a seam. Bollinger and Harrison [Bollinger 71] describe the principles and techniques of a spatial seam tracking system. In this application, the seam is constrained to lie on a cylindrical surface. Tomizuka, *et al.* [Tomizuka 80] propose a preview control strategy for two-axis welding torch positioning and velocity control. The scheme is only applicable for two-axis control and hence constrains the seam to lie in a plane. Furthermore, the scheme cannot be implemented on a general purpose six degree-of-freedom robot.

One of the first successful demonstrations of computer vision to arc welding is the NASA *weld skate* [Hill 80]. This system has been implemented on a special-purpose robot and requires special edge preparation to operate properly. The system is incapable of making any determination regarding the joint fit-up. The principle of structured illumination has been employed by Kawasaki Heavy Industries of Japan [Masaki 79] to develop a visual seam tracking system. The approach used for image analysis is *training-by-showing*. A set of

typical images is acquired in a teaching operation prior to welding and stored in the memory of the processor. As real-time images are acquired, a search is conducted (through the set of trained images) until a match is found. During the matching process, the positional displacement between the two images is computed and used to correct the position of the torch. The system has been designed specifically for the ship building industry and hence operates only on fillet joints. The system does not utilize part fit-up information for seam tracking.

An example of welding-by-teaching is the system described by Masaki, et al. [Masaki 81] which has a visual seam tracking capability. The robot is taught the reference path for the end-effector and the reference image for the image processor. In the teach mode, two passes are required for each work piece, one for sensing and one for welding. The path for the welding operation is generated from the sensing pass information.

Current robot welding systems are thus suitable for large batches of parts which are cut and fit to tight tolerances. The robot must trace and weld a seam, within acceptable tolerance limits, on such closely fit parts. Another limiting factor for a semi-automated welding system is the robot programming and set-up time in the shop when the part is changed. These constraints may be eased through the introduction of a CAD/CAM data base in which the welding trajectories and speed, weave pattern, wire feed rate, voltage and current are stored for each welding part and then retrieved as required. Unpredictable fit-up and loose part tolerances create the need for a real-time guidance and control algorithm.

The objective of this paper is to introduce a versatile seam tracing algorithm that demonstrates the feasibility of tracking a seam in real-time. The general purpose seam tracing algorithm can be implementabled on any robot with six degrees-of-freedom. The algorithm is motivated by the physical interpretation of the forward solution, or T_6 matrix [Paul 81] and the inverse Jacobian. To facilitate implementation of the inverse Jacobian solutions, an iterative algorithm is developed to compute the differential changes in the joint variables from the dT_6 matrix. To reduce significantly the on-line computational requirements, the concept of a modified dT_6 matrix is also introduced. To evaluate the performance of the seam tracing algorithm, a functional simulation package (for the Cyro[1] robot in our laboratory) has been implemented. The outputs of the simulation are the joint position and velocity set-points for the robot control system.

The paper is organized as follows. The kinematics of the Cyro robot (including the forward and reverse solutions , and the Jacobian and inverse Jacobian) are developed in Section 2. The foundations for the seam tracking algorithm are laid in Section 3. Focus is on the specification of the T_6 matrices at the sample points

---

[1]Cyro is a trademark of the Advanced Robotics Corporation.

along the seam and planning the motion of the torch. To reduce the computational requirements of the algorithm for real-time applications, the concept of the modified dT_6 matrix is introduced in Section 4, and an iterative algorithm to compute both large and small changes in the joint coordinates is then developed. The computational requirements are enumerated to indicate the potential for the real-time implementation of the algorithm. The salient features of the simulator, which has been implemented to evaluate the performance of the seam tracing algorithm, are presented in Section 5. Simulation experiments for representative test cases are then highlighted in Section 6. Finally, in Section 7, conclusions are drawn from the simulation experiments, and the paper is summarized.

2. Kinematics of the Cyro Robot

The forward solution (or T_6 matrix) of the robot, from the base frame to the torch (or end-effector) frame, is developed using homogeneous transformations [Paul 81]. The homogeneous transformations, relating two successive coordinate frames, are only a function of the six joint coordinates. Thus, knowledge of all of the ° joint coordinates leads to the transformation (or forward solution) from the base frame to the torch frame.

To develop the homogeneous transformation or A matrices, a coordinate frame is embedded in each of six links of the robot, using the Denavit-Hartenberg convention [Denavit 55]. The coordinate frames a. shown in Figure 1. Joints 1, 2 and 6 are revolute, and joints 3, 4 and 5 are prismatic. The coordinates of the revolute joints are θ_1, θ_2 and θ_6; and the coordinates of the prismatic joints are x_3, z_4 and y_5. The subscripts on the coordinates indicate the joint number; the base is link zero. The base coordinate frame is fixed at the center of the table of the robot and coincides with the first coordinate frame. When *all* of the six joint coordinates are zero, the axes for joint 1 (table) and joint 6 (torch) are parallel and the robot becomes singular. In the algorithm, the manipulator is assumed to be at the zero position. Without loss of generality, the constant offsets of the robot are assumed to be zero. A counter-clockwise rotation of the revolute joints is considered to be positive, and translation of the prismatic joints along the positive z-axis is considered to be positive.

The link parameters of the Cyro robot are listed in Table 1, and the forward solution is displayed in Table 2. Having obtained the forward solution, the values of the joint coordinates that led to the T_6 matrix can be computed. This reverse solution [Paul 81] is required (by the simulator) to relate the T_6 matrix to the present values of the joint coordinates. The reverse solution is listed in Table 3.

The differential changes in the cartesian coordinates of the torch are related to the differential changes in the joint coordinates through the manipulator Jacobian [Whitney 72, Paul 81]. Each column of the Jacobian matrix J is a differential translation and rotation vector. The column vectors of

$$J = \left[\frac{\partial T_6}{\partial \theta_1} \quad \frac{\partial T_6}{\partial \theta_2} \quad \frac{\partial T_6}{\partial x_3} \quad \frac{\partial T_6}{\partial z_4} \quad \frac{\partial T_6}{\partial y_5} \quad \frac{\partial T_6}{\partial \theta_6} \right]$$

are listed in Table 4.

In seam tracing, sensory data can be utilized to determine the incremental change in the position of the seam from differential changes in the elements of the T_6 matrix. The differential change matrix dT_6 is thus available to plan the incremental motions of the torch. It thus becomes imperative to find the inverse Jacobian (or incremental changes in the joint coordinates) which produce the specified incremental change in the T_6 matrix.

Numerical inversion of the Jacobian [Whitney 72] is computationally intensive and hence is not suitable for real-time control applications. Incremental changes in the joint coordinates can be obtained from a Taylor series expansion of the reverse solution. Such an approach leads to analytical formulae for the differential joint coordinates which are functions of the elements of the T_6 and dT_6 matrices [Paul 81]. Analytical formulae for the differential joint coordinates, which are obtained by differentiating the reverse solution (in Table 3), are listed in Table 5.

3. Kinematic Seam Tracking Control

The control task is to fill a volume with weld material while maintaining the proper position and orientation of the torch with respect to the seam. While traversing the seam, the tip of the torch traces a curve in 3-D space. If the discrete points on the curve to be traced and the surface containing the curve are identified, the T_6 matrices can be generated for each point on the discretized curve.

Specification of the T_6 matrices at the sample points of the discretized curve accomplishes the seam tracing task. The T_6 matrix is

$$T_6 = \begin{bmatrix} n & o & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and represents the position and orientation of the torch shown in Figure 2. The origin of the describing coordinate frame is located at the tip of the torch and is described by the vector p with respect to the base frame. The three unit vectors n, o and a, which describe the orientation relative to the base frame, are directed as follows [Paul 81]. The z-axis of the describing frame lies along the direction that the torch approaches the surface (containing the curve to be traced) and is called the approach vector a. The y-axis of the describing frame lies along the direction of the boom holding the camera and is called the orientation

vector **o**. The normal vector **n** is then chosen to form a right-handed set of vectors and is computed as

$$\vec{n} = \vec{o} \times \vec{a}$$

The vectors **n**, **o** and **a** describing the orientation of the torch and the vector **p** describing the position can be specified independently. The control task can thus be split into two independent components :

- Tracing a curve in 3-D space; and

- Maintaining proper orientation of the torch with respect to the surface which contains the curve to be traced.

The volume to be filled with the weld material is contained within two surfaces (of metal) which are to be joined together. The surfaces may be non-overlapping, as in the case of a butt joint (in Figure 3), or overlapping as in the case of a lap joint or a fillet joint.

Let m be a curve (in 3-D space) which lies on the surface S and is to be traced by the tip of the torch. The surface S may have a varying slope. Henceforth, the curve m will be termed the *mid-seam*. The mid-seam is discretized length-wise. Let m_i be the vector (with respect to the base frame) pointing to the i-th sample point m_i on the mid-seam. The discretization is specified to allow a piecewise linear approximation of the curve m between two adjacent sample points. If the surface (in the vicinity of the two sample points m_i and m_{i+1}) is also discretized, then a piecewise planar approximation of S is obtained. Let P_i denote the plane (containing the points m_i and m_{i+1}). The direction cosines of the plane specify the orientation of the torch which is held constant for the duration of travel from m_i to m_{i+1}.

The position of the torch is specified by the coordinates of the sample points. The foregoing description of the position and the orientation of the torch completes the formulation of the T_6 matrix at the sample points. The ensuing section specifies the T_6 matrix for butt, lap and fillet joints.

3.1. Specification of the T_6 Matrix

Having interpreted physically the elements of the T_6 matrix, the next step is to generate numerical values of the elements of the T_6 matrix in terms of the coordinates of the sample points obtained from the sensor system. Since a light stripe projector and a solid state camera are assumed to be used as the sensory device, the 3-D coordinates of the points on the surfaces to be joined are mapped into pixels in the camera image. Figure 4 shows a typical camera image (at the i-th sampling instant) obtained from a butt joint. The break points in the camera images of the surface indicate the discontinuity in the actual surfaces to be welded. To specify the curve to be traced, it is essential to extract 3-D coordinates of the break points in the images. The

information regarding the orientation of the surfaces, in the vicinity of the break points, can be obtained by extracting the coordinates of one additional point on each surface. The break points are called u_i and v_i for the butt joint shown in Figure 3. The additional points on each of the two surfaces are called p_i and q_i, respectively. The subscript i denotes the sampling instant. The coordinates of the sample points are specified with respect to the base frame of the robot, and p_i, u_i, v_i and q_i are the vectors from the origin of the base frame to the points p_i, u_i, v_i and q_i respectively. The edges formed by the sample points $\{\,p_i\,\}$, $\{\,q_i\,\}$, $\{\,u_i\,\}$ and $\{\,v_i\,\}$ are denoted by p, q, u, and v, respectively.

To specify the p vector of the T_6 matrix requires knowledge of the sample points along the mid-seam. For a butt joint, the requirement that the torch be placed exactly in the middle across the u and v edges forces the x, y and z coordinates of the mid-seam to be computed as:

$$
\begin{aligned}
m_{xi} &= \frac{u_{xi} + v_{xi}}{2} \\
m_{yi} &= \frac{u_{yi} + v_{yi}}{2} \\
m_{zi} &= \frac{u_{zi} + v_{zi}}{2}
\end{aligned}
$$

(1)

For both lap and fillet joints, torch stand-off is an important consideration for obtaining a quality weld. Let the desired torch stand-off be characterized by the parameter s, where s ranges from 0 to 1. The coordinates of the mid-seam are then computed as

$$
\begin{aligned}
m_{xi} &= u_{xi} + s(v_{xi} - u_{xi}) \\
m_{yi} &= u_{yi} + s(v_{yi} - u_{yi}) \\
m_{zi} &= u_{zi} + s(v_{zi} - u_{zi})
\end{aligned}
$$

(2)

The p_x, p_y and p_z components of the T_6 matrix are

$$p_x = m_{xi} \tag{3}$$
$$p_y = m_{yi} \tag{4}$$
$$p_z = m_{zi} \tag{5}$$

and specify completely the last column of the T_6 matrix.

Practical seams have edges with slowly-varying slopes. Since the sample points are assumed to lie close to each other (typically separated by 1 mm), the edges between two sample points can be approximated by a

straight line. This approximation leads to a plane which passes through at least three of the four sample points (p_i, p_{i+1}, q_i, q_{i+1}). The direction cosines of the plane are also the direction cosines of a vector normal to the plane. Since the torch is required to be perpendicular to the *fictitious* seam surface, the direction cosines of the approach vector a are specified as the *negative* of the direction cosines of the normal to the plane.

The values of the components of the orientation vector o are computed under the following constraint. When the torch and the camera are on the seam, both should track the seam. This condition guarantees that sample points will not be lost if the slope of the mid-edge (at the point of the torch) differs from the slope at the point-of-view of the camera. Let M be a constant shift in the number of sample points between the torch and the camera. Figure 2 shows that the o vector is perpendicular to the a vector and points along the direction of the arm holding the camera. If the camera is traveling in the direction of the line joining m_{i+N-1} and m_{i+N}, then the robot system will never lose track of the seam, unless the slope experiences large changes or discontinuities along the seam.

The equation of a plane passing through three of the four points (p_{i+N}, p_{i+N-1}, q_{i+N}, q_{i+N-1}) can now be specified, and the angle between the planes at the camera and the torch points can be determined. If (λ_1, μ_1, ν_1) and (λ_2, μ_2, ν_2) are the direction cosines of the two planes, then the cosine of angle θ between the planes is

$$\cos(\theta) = \lambda_1\lambda_2 + \mu_1\mu_2 + \nu_1\nu_2$$

(6)

The vector α joining the points m_{i+N-1} and m_{i+N} along the mid-edge is computed as

$$\vec{\alpha} = \vec{m}_{i+N} - \vec{m}_{i+N-1}$$

(7)

The projection of the vector α onto the torch plane is chosen to be the o vector and is normalized to be of unit length. Having obtained the normalized o and a vectors, the n vector is computed as

$$\vec{n} = \vec{o} \times \vec{a}$$

(8)

to specify completely the T_6 matrix at the sample points.

While traversing from the i-th to the (i+1)-th point, the orientation of the torch and consequently the n, o and a vectors remain constant. The p vector in the T_6 matrix changes linearly because of the straight line approximation between the two points. To maintain a continuous speed and acceleration at the end points of

the segment, the motion of the torch is planned.

3.2. Planning the Motion of the Torch

The motion of the tip of the torch, in traversing a segment, is composed of two parts:

- Motion along the segment, and

- Transition between segments.

To make a smooth transition between segments, it is desirable to maintain a continuous velocity and acceleration at the transition points. To specify the transition equations, a fourth-order curve is fit between the point where a transition starts and the point indicating the end of the transition [Paul 81]. The equations for the transition trajectories and velocities are outlined in Section 3.3.

Let M be the total number of transition steps in which the desired change in T_6 is to occur. The differential changes in the n, o and a vectors at each transition point are

$$\Delta \vec{n} = \frac{\vec{n}_{(i+1)} - \vec{n}_i}{M}$$

(9)

$$\Delta \vec{a} = \frac{\vec{a}_{(i+1)} - \vec{a}_i}{M}$$

(10)

$$\Delta \vec{o} = \frac{\vec{o}_{(i+1)} - \vec{o}_i}{M}$$

(11)

where the subscripts i and (i+1) denote the i-th and (i+1)-th segments, respectively. The Δn, Δo and Δa vectors are then added to the current n, o and a vectors, respectively, and normalized to unit length, to produce the n, o and a vectors at the next segment. Let n_p, o_p, a_p and n_n, o_n, a_n be the vectors of the T_6 matrix at the present and next segments, respectively; and let the subscript N indicate that these vectors have been normalized to unit length. The n_n, o_n and a_n vectors are thus computed as follows :

$$\vec{n}_n = (\vec{n}_p + \vec{\Delta n})_N$$

(12)

$$\vec{o}_n = (\vec{o}_p + \vec{\Delta o})_N$$

(13)

$$\vec{a}_n = (\vec{a}_p + \vec{\Delta a})_N$$

(14)

Having computed the three component vectors (n, o and a) of T_6 at the next segment, the first three columns of the differential change matrix dT_6 are determined, and the fourth column (or dp vector) can be computed from the transition trajectory to specify the differential translation. The differential change matrix dT_6 is then

$$dT_6 = \begin{bmatrix} dn_x & do_x & da_x & dp_x \\ dn_y & do_y & da_y & dp_y \\ dn_z & do_z & da_z & dp_z \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

(15)

where the differential vector components (dn, do, da and dp) of the dT_6 matrix represent the corresponding change in the vector components (n, o, a and p) of the T_6 matrix.

Having generated the present T_6 matrix and dT_6 matrix to reach the next point, the inverse Jacobian (in Table 5) is used to compute *iteratively* the differential changes in the joint coordinates. The inverse Jacobian solutions are derived under the assumption that the changes in the joint variables leading to the specified dT_6 matrix are *small*. To overcome the practical fact that this assumption is *not* always satisfied in seam tracing applications, an iterative technique is developed to compute the changes in the joint variables. The velocity set-points during the transition are obtained by dividing the incremental values of the joint variables by the time required to make the incremental change. The transition ends when the torch reaches the point D on the segment B-C in Figure 5. At this point, the torch has the required orientation and velocity to track the (i + 1)-th segment D-C without error. This motion is called *motion-along-the-segment*.

During the motion of the torch, along the segment, the n, o and a vectors of the robot remain constant. The updated dT_6 matrix is specified as:

$$dT_6 = \begin{bmatrix} 0 & 0 & 0 & dp_x \\ 0 & 0 & 0 & dp_y \\ 0 & 0 & 0 & dp_z \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

(16)

where dp_x, dp_y and dp_z are the differential changes in the x, y and z coordinates of the (i + 1)-th transition

point. When the torch reaches the next transition point, the process of planning the motion of the torch, along the segment and during the transition, is repeated to plan the motion for the next segment.

3.3. The Transition Curves

A transition segment is illustrated in Figure 5. The transition starts at point A in the i-th segment and ends at point D in the (i+1)-th segment. Maintaining a continuous velocity and acceleration at the points A and D appears to require that six boundary conditions be satisfied. A fifth-order polynomial (with six parameters) would then be needed to approximate the cartesian transition curve. *Symmetry* of the transition guarantees that a *quartic* polynomial can approximate the cartesian transition curve [Paul 81][2]. To facilitate the development, let τ be the transition time and T be the time required to traverse the segment B-C. The time of travel (T) across a segment is computed by dividing the volume of the weld material (to be deposited along the segment) by the weld-wire volumetric feed-rate, which is assumed to be constant. The ratio (τ/T) is specified by the engineer (in Section 6). Let the *normalized* time-step parameter h be defined according to

$$h = \frac{t + \tau}{2\tau}$$

(17)

where t denotes the running time-variable $(-\tau < t < \tau)$.

Let the fourth-order polynomial approximating the cartesian transition segment be

$$X(h) = \beta_4 h^4 + \beta_3 h^3 + \beta_2 h^2 + \beta_1 h + \beta_0$$

where the five parameters $(\beta_i$ for $i = 0$ to 4) must be selected for X(h) to satisfy the boundary conditions [Paul 81]:

$$X(0) = A ; \qquad X(1) = (C-B)\frac{\tau}{T} + B$$

(18)

$$\dot{X}(0) = 2(B-A) ; \qquad \dot{X}(1) = 2(C-B)\frac{\tau}{T}$$

(19)

$$\ddot{X}(0) = 0 ; \qquad \ddot{X}(1) = 0$$

(20)

where the *dot* denotes differentiation with respect to h. (The initial acceleration condition $\ddot{X}(0) = 0$ leads to $\beta_2 = 0$.)

The cartesian position, velocity and acceleration of the torch on the transition curve (as functions of the normalized time-step parameter h) are

---

[2] The transition equations are reproduced here because of the typographical errors in the cited reference.

$$X(h) = -(\Delta C \frac{T}{T} - \Delta B)h^4 + 2(\Delta C \frac{T}{T} - \Delta B)h^3 + 2(\Delta B)h + A$$

$$\dot{X}(h) = -4(\Delta C \frac{T}{T} - \Delta B)h^3 + 6(\Delta C \frac{T}{T} - \Delta B)h^2 + 2(\Delta B)$$

(21)

(22)

and

$$\ddot{X}(h) = -12(\Delta C \frac{T}{T} - \Delta B)h^2 + 12(\Delta C \frac{T}{T} - \Delta B)h$$

(23)

where $\Delta B = (B - A)$ and $\Delta C = (C - B)$.

Equation (21) defines the cartesian position of the torch (in terms of the normalized time-step parameter h) and is used to evaluate the coordinates of the transition point. The welding torch transits from the present segment to the next and maintains a continuous velocity and acceleration at the end points (A and D in Figure 5) of the transition segments. Equation (21) is used to compute the position of the torch during the transition. In the next section, the n, o and a vectors for the transition are formulated.

4. The Modified dT$_6$ Matrix

The dT$_6$ matrix specifies an incremental change in the orientation and position of the torch induced by incremental changes in the joint coordinates. Seam tracing requires incremental changes (in the base coordinates) in both the position and orientation of the torch. Since the changes in the base and the joint coordinates are related through the nonlinear inverse Jacobian coordinate transformation, a small change in the position and orientation of the torch in the base coordinates may require a large change in the joint variables. This realization hampers application of the inverse Jacobian to compute the differential changes in the joint coordinates from the dT$_6$ matrix.

The goal of this section is to introduce the concept of a modified dT$_6$ matrix and an iterative algorithm which does not restrict the nature of changes in the joint variables that led to the specified dT$_6$ matrix [Khosla 83]. If the changes are incremental (as assumed for the derivation of the inverse Jacobian), then the algorithm converges in the first iteration. In the case of large changes, the algorithm converges rapidly (in typically 2-3 iterations for the examples highlighted in Section 6) to the appropriate differential changes in the joint coordinates. The T$_6$ matrix at the next point is computed by adding the present T$_6$ matrix to the modified dT$_6$ matrix. This approach reduces computation time because computing the next T$_6$ matrix does not require the forward solution.

Let R and S be two points on the mid-seam transition segment between the points A and D (in Figure 6).

Let the position and the orientation of the torch at points R and S be specified by T_{6R} and T_{6S}, respectively. Let T_{6S} be such that the change dT_6 in T_{6R} leads to T_{6S}. Thus,

$$T_{6S} = T_{6R} + dT_6$$

(24)

It is also possible to reach T_{6S} from T_{6R} through a transformation C (in the base coordinates) which consists of a translation along the x, y and z axes followed by a rotation θ along an axis k. Thus,

$$T_{6S} = CT_{6R}$$

(25)

where

$$C = Trans(x,y,z) Rot(k,\theta)$$

(26)

Let n_R, o_R and a_R be the component vectors of the T_{6R} matrix specifying the orientation of the torch at the point R, and n_S, o_S and a_S be the corresponding vectors of the T_{6S} matrix. From (24), the corresponding differential vectors dn, do and da are

$$\vec{dn} = \vec{n_S} - \vec{n_R}$$

(27)

$$\vec{do} = \vec{o_S} - \vec{o_R}$$

(28)

$$\vec{da} = \vec{a_S} - \vec{a_R}$$

(29)

While traveling from point A to point D (in Figure 5), the dT_6 matrices should be computed to preserve the physical significance of the T_6 matrices at the transition points. Thus, T_{6R} should be computed from dT_6 according to (24). Geometrically, the loci of the n, o and a vectors of a T_6 matrix should be a sphere of unit radius.

Figure 6 depicts the total desired change in each of the n, o and a vectors, while traversing from point A to point D (in Figure 5) in M steps. The components Δn, Δo and Δa, computed from (30)-(32), are shown in Figure 6. The arc of the unit circle represents the loci of the n, o and a vectors during the transition. Let $(n_R)_u$, $(o_R)_u$ and $(a_R)_u$ be the unnormalized vectors whose tips lie on the point R' on the straight line joining points A and D. The subscript u indicates that the vectors are unnormalized. The vectors n_R, o_R and a_R are obtained by normalizing the magnitude of the corresponding vectors to unit length.

The unnormalized vectors of the T_{6S} matrix are computed as

$$(n_s)_u = (n_r)_u + \delta n$$

(30)

$$(o_s)_u = (o_r)_u + \delta o$$

(31)

$$(a_s)_u = (a_r)_u + \delta a$$

(32)

and then normalized to obtain the n_S, o_S and a_S vectors. The right-hand sides of (27)-(29) are thereby specified completely, and the differential vectors of the dT_6 matrix can be computed to preserve the physical significance of the T_6 matrix.

Having obtained the dn, do and da vectors of the dT_6 matrix in (15), it remains to compute the dp vector to specify completely the matrix. The vector dp is computed as

$$\vec{dp} = \vec{p}_S - \vec{p}_R$$

where the vectors p_R and p_S are obtained from (21).

By construction, the dT_6 matrix satisfies (26). Since the magnitudes of the changes in the joint variables are *not* constrained in the derivation of the dT_6 matrix, the computed dT_6 matrix satisfies (27). Thus, the T_6 matrix at the next point S can be obtained by adding the T_6 matrix at the present point R to the computed dT_6 matrix (and consequently there is no need to compute the T_6 matrix at the point S from the updated values of the joint variables when the torch reaches the point S). The real-time computational requirements of this construction are detailed in Section 5.

4.1. An Iterative Algorithm to Compute the Differential Changes

In applications, such as seam tracing (in which the sample points lie at incremental distances along the seam), the required changes in the joint variables may not be incremental. A practical example involves tracing a seam which has large slope variations. The dT_6 matrix is related to the changes in the joint variables through the inverse Jacobian which, in turn, is derived under the assumption of small changes in the joint variables. Many of the seams occurring in practice have slowly varying slopes and application of the inverse Jacobian to make the incremental motions is computationally advantageous. In practice, the seam may exhibit large slope changes at a few points, and the solution obtained (for the differential changes in the joint coordinates to reach the next point) from the inverse Jacobian may exhibit significant errors. The torch is thus placed at the incorrect point on the seam, and a large error (in the position and orientation of the torch) is introduced. To overcome this problem, an iterative algorithm is introduced. A block-diagram of the

iterative algorithm is shown in Figure 7.

The algorithm begins at $i = 0$ with the initialization of the following variables: T_6^* (T_6 matrix of the next point), T_{6i} (current T_6 matrix), and q_{i+1} (current values of the joint variables).

The algorithm implemented at the i-th iteration is :

$$dT_6 = T_6^* - T_{6i}$$

(33)

$$dq_{i+1} = J^{-1}[dT_{6i}, q_{i+1}]$$

(34)

$$q_{i+1} = q_i + dq_{i+1}$$

(35)

$$T_{6i} = F[q_{i+1}]$$

(36)

$$q_i = D[q_{i+1}]$$

(37)

where J^{-1} is the inverse Jacobian (in Table 5) for the computation of the differential changes in the joint variables from dT_{6i} and q_{i+1} (in contrast to the symbolic or numerical inversion of the Jacobian matrix); F denotes the operation of computing the T_6 matrix (in Table 2); and D signifies the computation (delay) time for the forward solution.

Upon substituting (34) into (35),

$$q_{i+1} = q_i + J^{-1}[dT_{6i}, q_{i+1}]$$

(38)

and hence,

$$dq_{i+1} = (q_{i+1} - q_i) = J^{-1}[dT_{6i}, q_{i+1}]$$

(39)

Equation (39), and consequently the algorithm depicted in Figure 7, is the Newton-Raphson method [Atkinson 78] for solving the inverse Jacobian system of nonlinear equations (in Table 5).

The algorithm converges (in theory) when *all* of the components of the dT_{6i} matrix in the block-diagram (in Figure 7) are zero. In practice, the algorithm is assumed to converge when each of the elements of dT_{6i} is less than a pre-set tolerance. The computed vector q_i contains the desired set-points (in the joint variables) to reach the next position. When the desired changes in the joint variables are small, the algorithm converges in one iteration and reduces to obtaining the inverse differential solutions from the inverse Jacobian (in Table 5).

For the seams tested with the simulator, the algorithm always converged in a maximum of three iterations (when the preset tolerance was set equal to zero). The computational requirements of the seam tracing algorithm are outlined in the next section.

5. Computational Requirements

The computations required to follow the seam from the present sample point to the next are outlined in Table 6. The number of iterations required for the transition point computation and iterative algorithm (33)-(37) are denoted by N_1 and N_2, respectively. (For the seams traced by the simulator, typical values are found to be $N_1 = 2$ and $N_2 = 1$.) Execution times of the 8087 hardware instructions [Intel 83] are listed in Table 7. These floating point operation times (including the times required to load and store the operand) are used to estimate the time required to move the torch from one sample point to the next. Typical times (shown in Table 6) range from 24 to 29 milliseconds, which correspond to sampling rates of 35-40 Hz. For most welding applications, a sampling frequency of 10 Hz appears to be adequate.

These computational estimates are based upon the *matrix* kinematic modeling of manipulators used throughout this paper. Matrix representations of rotations are highly redundant. *Quaternions* [Beeler 72, Hamilton 69] offer a convenient representation for rotations and can reduce both the storage requirements and computational load [Taylor 79]. The authors estimate that the quaternion implementation of the seam tracking algorithm would increase the achievable sampling rate to 60 Hz.

6. Simulation

To evaluate the algorithm, a software simulator has been developed (in the C programming language on a VAX 11/780) for the six degree-of-freedom Cyro robot in our laboratory. The simulation is initialized by retrieving seam data (as coordinates of sample points) from a data file. The first two sampled cross-sections of the seam are used to compute the desired T_6 matrix of the robot at the first point on the mid-seam and to compute the joint position and velocity set-points to reach the desired destination. Upon reaching the first point on the mid-seam, the algorithm computes the coordinates of the transition point on this segment (for the ratio τ/T which is entered by the engineer at the start of the simulation) and the total time T required to traverse the next segment. The T_6 matrices at the present and next sample points are used to generate the joint position and velocity set points by the algorithm in (33)-(37). The trajectory from the beginning to end of the transition is computed from (21), and the joint position and velocity set-points are computed to follow the interpolated curve. Upon reaching the end of the transition, the process is repeated (for each transition point), until the last sample point is reached. The simulation is then terminated and the specified curve is traced.

To achieve a quality weld, the lag or lead angle of the torch must be controlled adaptively. In the case of fillet and lap joints, the torch stand-off must also be controlled. Since an adaptive controller for these parameters remains to be developed, we have included (in our simulator) the facility to specify these control parameters at the beginning of the simulation.

In the software simulator, the torch can be rotated about *two* axes. The first is about an axis *parallel* to the direction of travel and the second is about an axis *perpendicular* to the direction of travel (in the plane of the first axis). This capability allows control over the lead or lag angle of the torch. Facility to specify the stand-off for lap and fillet joints has also been incorporated. The simulator has tracked butt, lap and fillet joints and the experimental results are highlighted in the next section.

7. Experimental Results

The seam tracking algorithm approximates the curve between sample points by a straight-line. To emphasize the effect of linear interpolation and the choice of sampling distance on the tracking accuracy, the simulator tracked a sinusoidal curve (with uniform spacings of 30 and 10 degrees). The simulation results are depicted in Figures 8 and 9, respectively. As the sampling distance is *decreased*, the tracked curve approaches the actual curve. The maximum tracking error (which occurs for h = 0.5) is

$$\epsilon = \frac{3}{16}\Delta C \frac{\tau}{T} - \frac{3}{16}\Delta B$$

and depends upon ΔB, ΔC and the ratio τ/T. For a particular seam and sample points, ΔB and ΔC are constant and the tracking error (with respect to the interpolated curve) is a *linear* function of the ratio τ/T. The simulation experiments illustrate that accurate tracking can be achieved by judiciously selecting the sample points and maintaining the ratio (τ/T) of transition time to segment time as small as possible.

8. Conclusions

A general-purpose real-time seam tracing algorithm, for implementation on any six degree-of-freedom robot, is proposed. The algorithm (which requires knowledge of only one-point-ahead to track a seam) can be applied to a multitude of robotic seam tracking activities such as gluing, surface grinding and flame cutting. The algorithm incorporates the physical interpretation of the T_6 and dT_6 matrices to realize seam tracking. To reduce the computational requirements, the paper introduces the concept of a modified dT_6 matrix. The inverse Jacobian solution is generalized (according to Newton's method) to compute both large and small changes in the joint coordinates.

To test the efficacy of the proposed seam tracing algorithm, a simulator has been written and tested on a VAX 11/780. The simulation results are highlighted in Section 7 of [Khosla 83]. The tracking accuracy is a function of the sampling distance because of the straight-line approximation between two successive sample

points, and the tracking error increases with the increase in ratio of the transition time to the segment travel time.

Future activity will focus on the adaptive control of the weld parameters and dynamic robot control. Successful practical implementation will depend upon the availability of faster processors and the experimental performance evaluation of the algorithm.

References

[Agin 82]   G. J. Agin and P.T. Highnam.
                A Movable Light Stripe Sensor for Obtaining Three Dimensional Coordinate Measurments.
                In *Proceedings of SPIE: The Society of Optical Engineers 360*, pages 326-333. San Diego,
                CA. August 21-27, 1982.

[Atkinson 78]   K. E. Atkinson.
                An Introduction to Numerical Analysis.
                John Wiley and Sons, New York, 1978.

[Beeler 72]   M. Beeler, R. W. Gosper and R. Schroeppel.
                "Hakmem".
                AI Memo 239, Artificial Intelligence Laboratory, Massachusetts Institute of Technology,
                Cambridge. MA 02139, February, 1972.

[Bollinger 71] J. G. Bollinger and H. L. Harrison.
                Automated Welding Using Spacial Seam Tracings.
                Welding Journal 50:787-792, November, 1971.

[Denavit 55]   J. Denavit and R. S. Hartenberg.
                A Kinematic Notation for Lower Pair Mechanisms Based on Matrices.
                Journal of Applied Mechanics 77:215-221, June, 1955.

[Hamilton 69]   W. R. Hamilton.
                Elements of Quaternions.
                Chelsea Publishing Company, New York, 1969.

[Hill 80]   J. W. Hill, J. Kremers and D. Nitzan.
                Advanced Automation for Ship Building.
                Final Report. Department of Navy Contract N00024-80-C-2026, SRI International, Menlo
                Park, CA 94025, November, 1980.

[Intel 83]   *iAPX 86/88, 186/188 User's Manual, Hardware Reference*
                Intel Corporation, 3065 Bowers Avenue, Santa Clara, CA 95051, 1983.

[Khosla 83]   P. K. Khosla.
                Simulation of a Sensor Based Robotic Spatial Seam Tracing System.
                Master's thesis, Department of Electrical and Computer Engineering, Carnegie-Mellon
                University, Pittsburgh, PA 15213, 1983.

[Masaki 79]   I. Masaki.
                 Kawasaki Pais Memo No. 7904, Pais Center, Robot Group.
                 Technical Report, Kawasaki Heavy Industries Ltd., Japan, April, 1979.

[Masaki 81]   I. Masaki, et al.
                 Arc Welding Robot with Vision.
                 In *Eleventh International Symposium on Industrial Robots,* pages 813-817. Tokyo, Japan,
                     October 7-9, 1981.

[Paul 81]   R. P. Paul.
                 Robot Manipulators : Mathematics, Programming and Control.
                 MIT Press, Cambridge, MA, 1981.

[Taylor 79]   R. H. Taylor.
                 Planning and Execution of Straight Line Manipulator Trajectories.
                 IBM Journal of Research and Development 23(4):424-436, July, 1979.

[Tomizuka 80]   M. Tomizuka, D. Dornfeld and M. Porcell.
                 Application of Microcomputers to Automatic Weld Quality Control.
                 Journal of Dynamic Systems, Measurement, and Control 102(2):62-68, June, 1980.

[Whitney 72]   D. E. Whitney.
                 The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators.
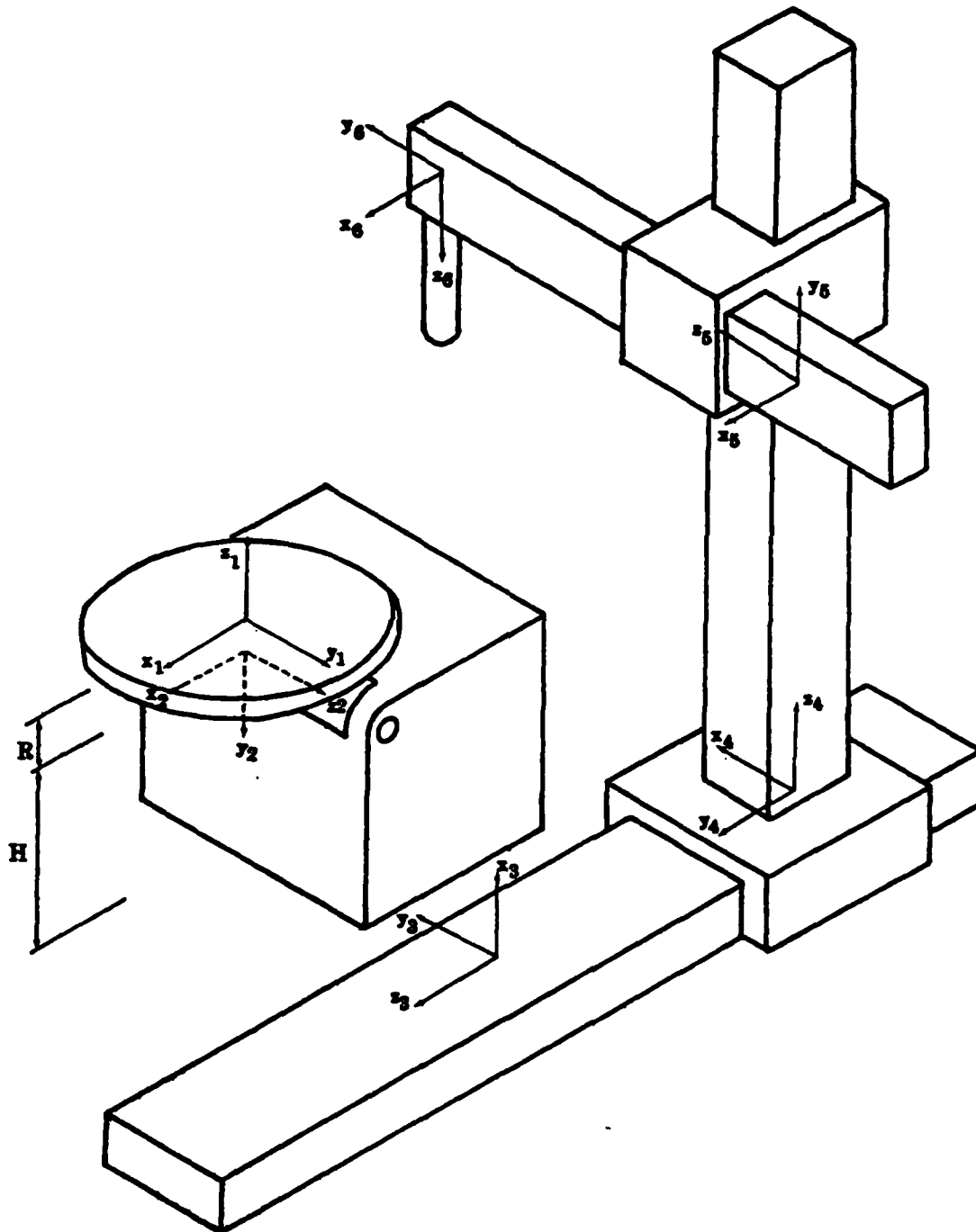   *Journal of Dynamic Systems, Measurement, and Control* 94(4):303-309, December, 1972.

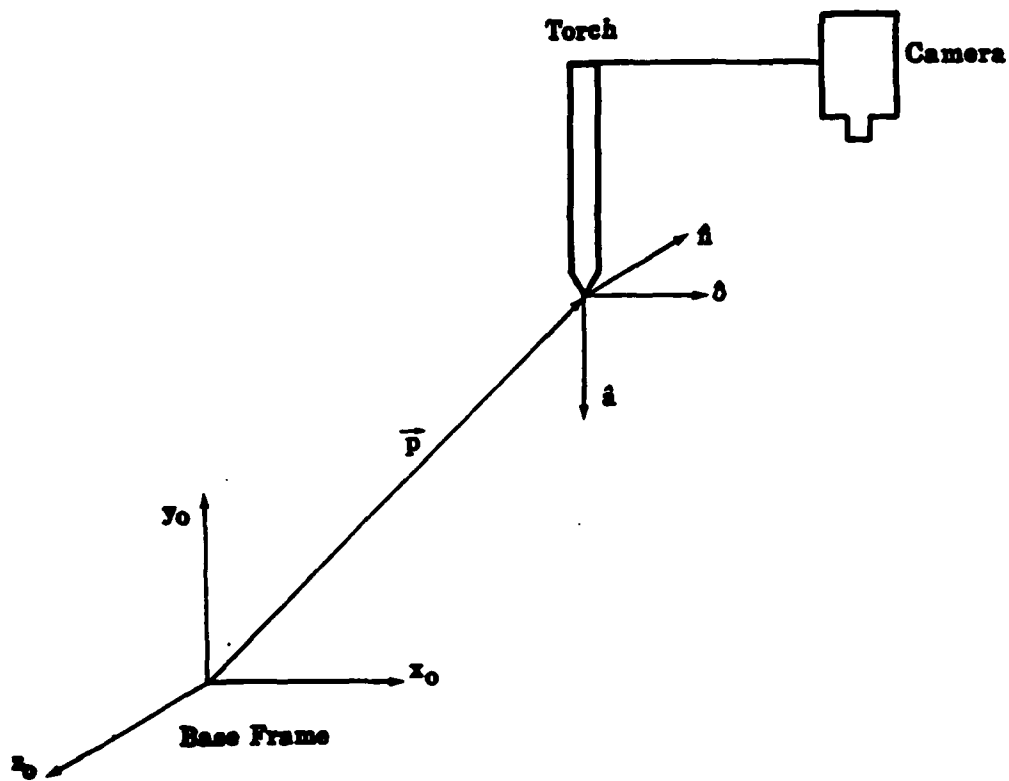**Figure 1:** Link Coordinate Frames of the CYRO Robot

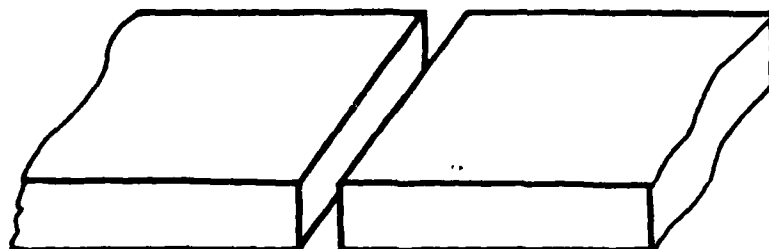**Figure 2:** Physical Interpretation of the T_6 Matrix
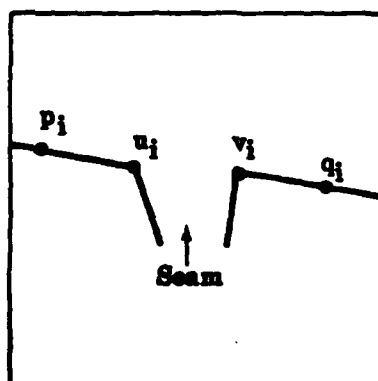
Figure 3: Butt Joint

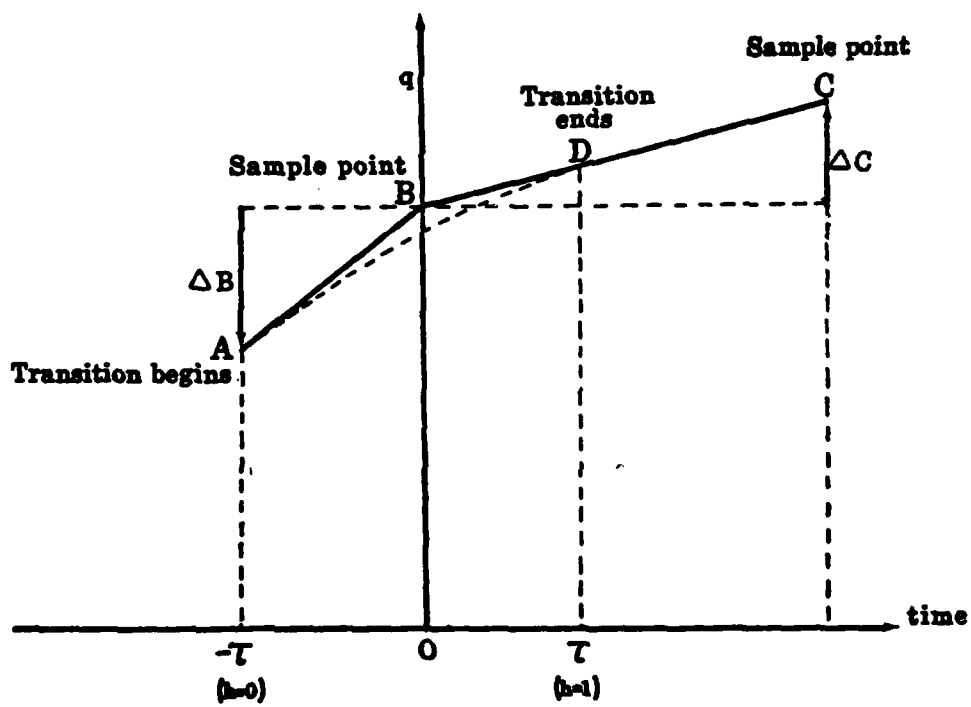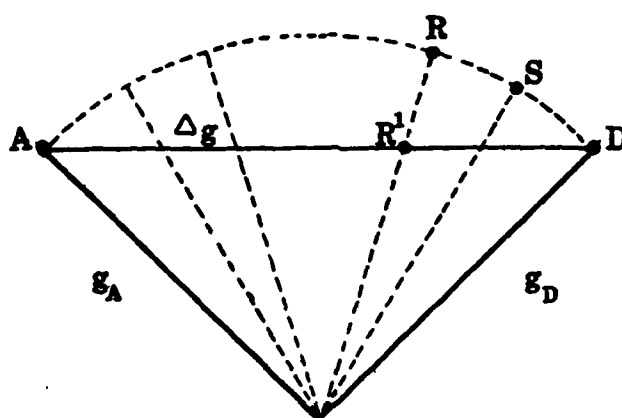Figure 4: Camera Image of a Butt Joint

Figure 5: A Transition Segment

Note· The generic vector g symbolizes the
normal (n), orientation (o) and approach
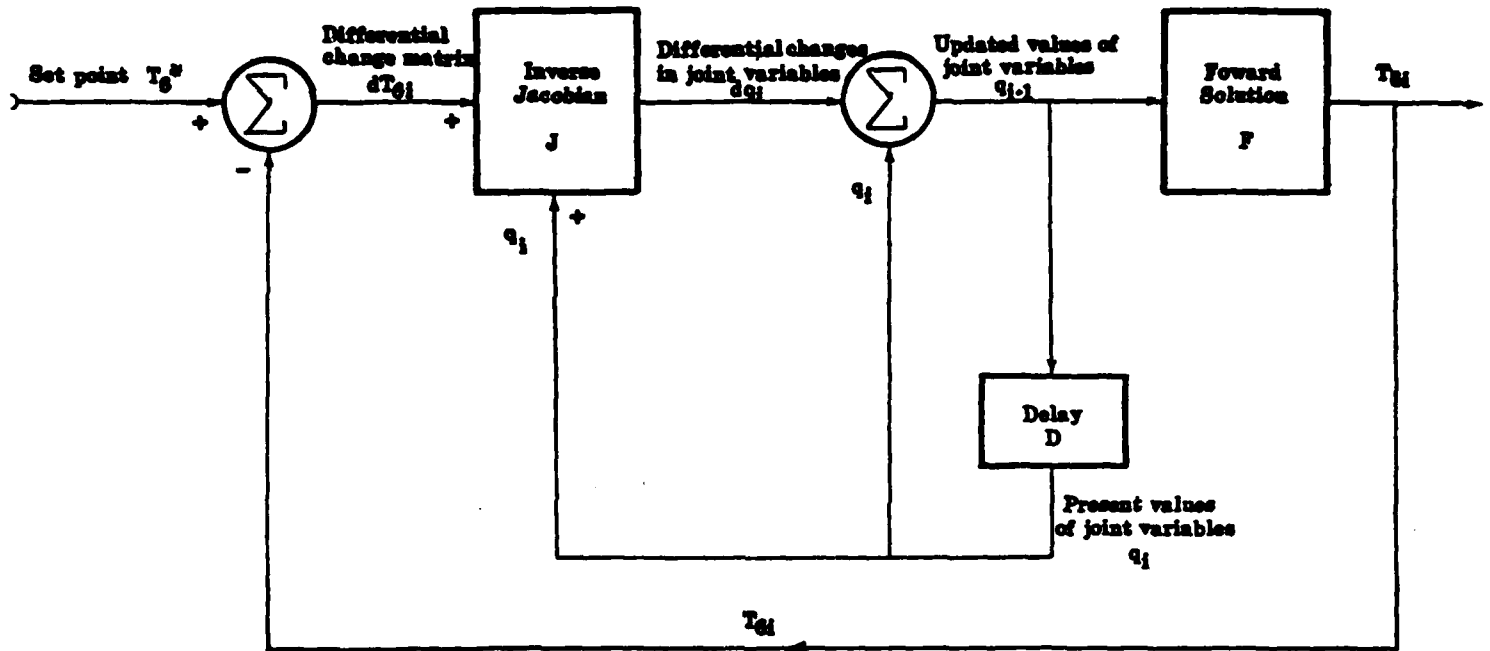(a) vectors.

Figure 6:   Locii of n, o and a Vectors

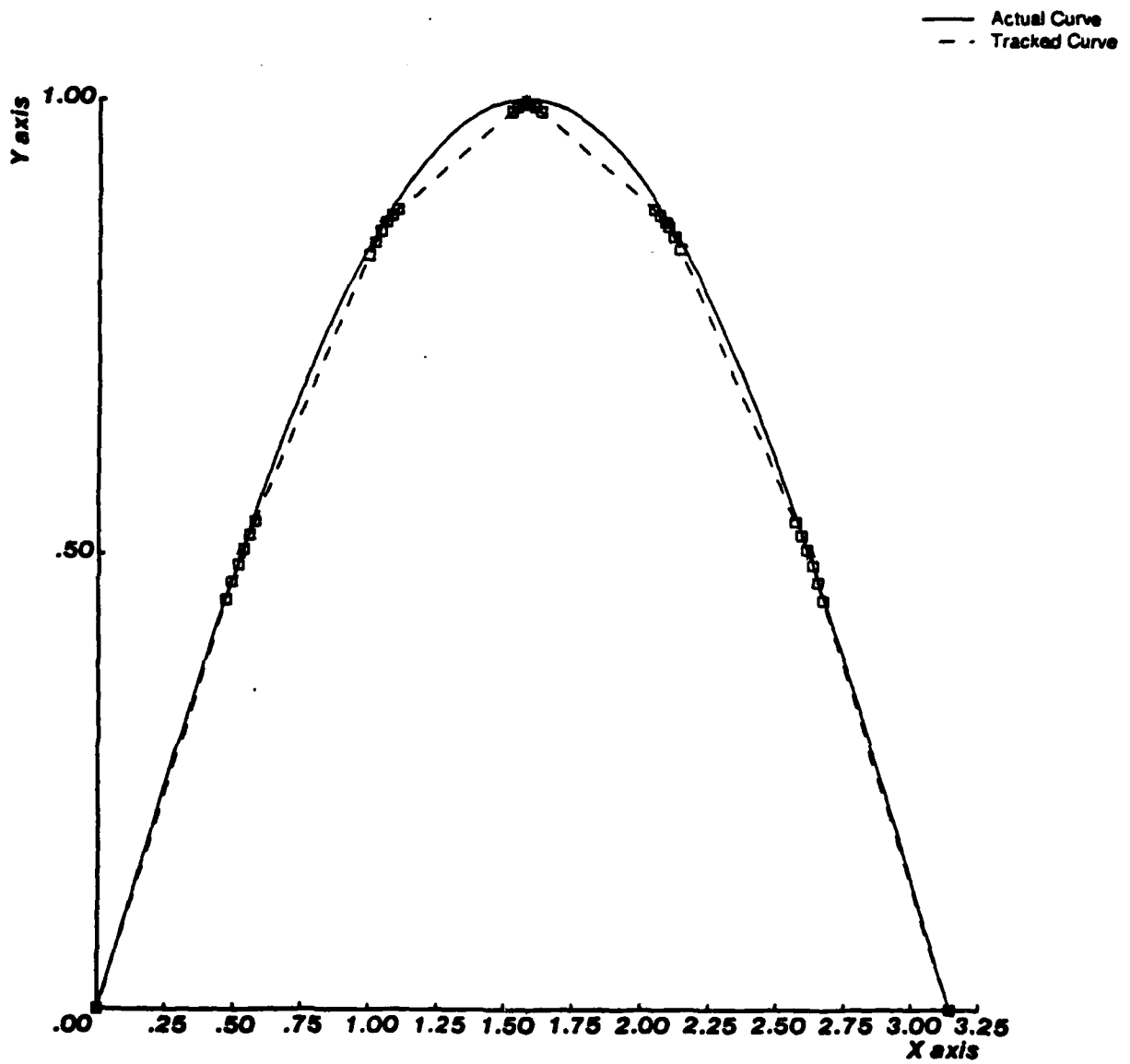Figure 7: Block Diagram of the Iterative Algorithm

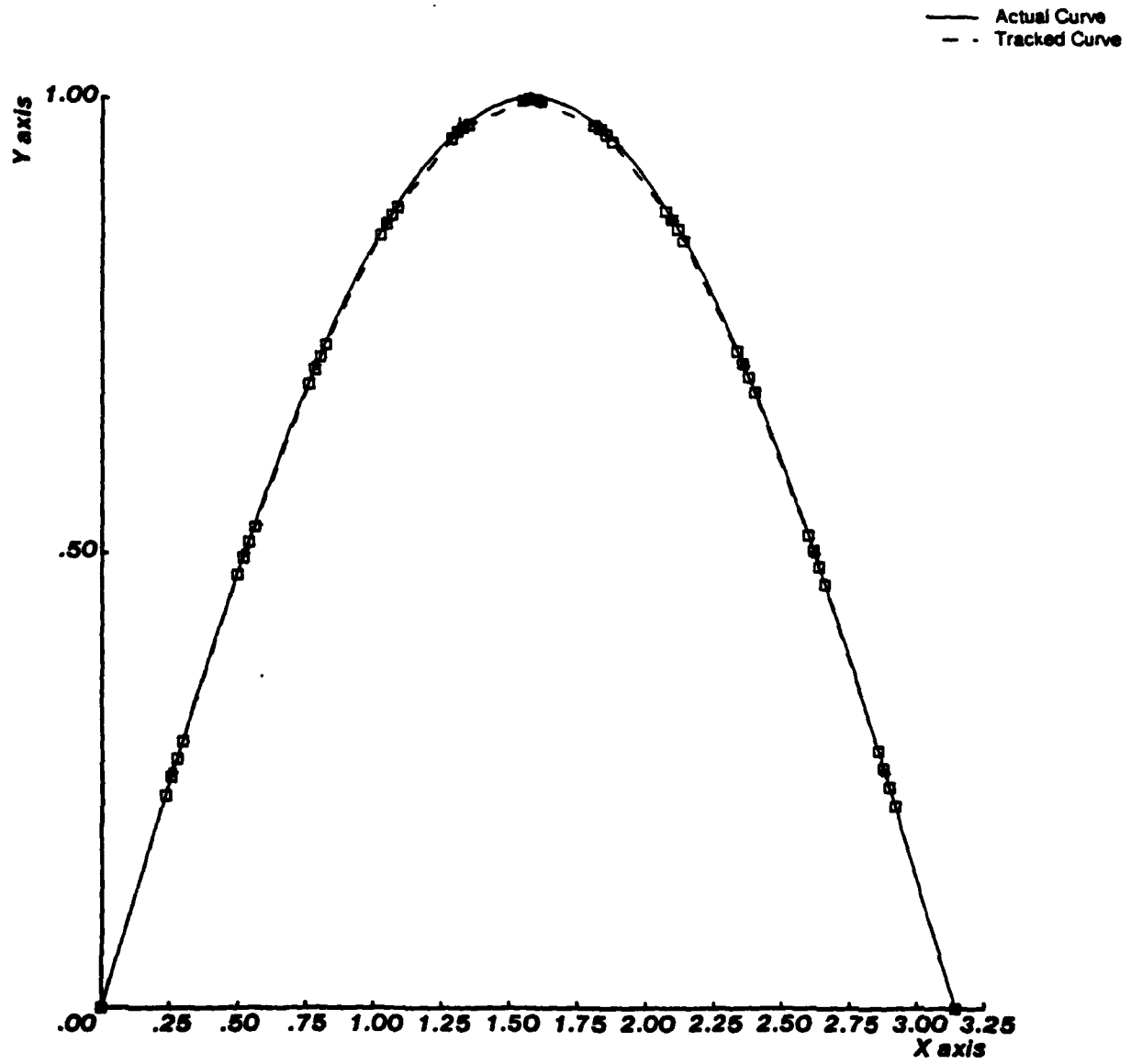**Figure 8:** Sinusoidal Seam Tracking (30 degrees sampling interval)

Figure 9: Sinusoidal Seam Tracking (10 degrees sampling interval)

Link	Variable	θ	α	a	d
1	θ_1	θ_1	-90	0	-R.
2	θ_2	$\theta_2 - 90$	-90	-H	L
3	x_3	90	90	0	x_3
4	z_4	90	90	0	z_4
5	y_5	0	0	0	y_5
6	θ_6	θ_6	0	0	0

> Definition of the Parameters
>
> θ is the angle of rotation about the z-axis
>
> α is the angle of rotation about the x-axis
>
> a is the length of translation along the x-axis
>
> d is the length of translation along the z-axis

Table 1: Link Parameters of the CYRO Robot

$$^0T_6 = \begin{bmatrix} C_1C_2C_6 + S_1S_6 & -C_1C_2S_6 + S_1C_6 & -C_1S_2 & C_1(z_4S_2 + x_3C_2 - HS_2) - S_1(-y_5 + L) \\ S_1C_2C_6 - C_1S_6 & -S_1C_2S_6 - C_1C_6 & -S_1S_2 & S_1(z_4S_2 + x_3C_2 - HS_2) + C_1(-y_5 + L) \\ -S_2C_6 & S_2S_6 & -C_2 & (z_4C_2 - x_3S_2 - HC_2) - R \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Table 2: Forward Solution of the CYRO Robot

Joint coordinate	Analytical Expression	Conditions
θ_1	$atan2\left[\dfrac{a_y}{a_x}\right]$	$T_6(2,3) > 0$
θ_1	$atan2\left[\dfrac{-a_y}{-a_x}\right]$	$T_6(2,3) < 0$
θ_2	$atan2\left[\dfrac{-(C_1a_x + S_1a_y)}{(-a_z)}\right]$	
x_3	$C_1C_2p_x + S_1C_2p_y - S_2p_z - S_2R$	
z_4	$C_1S_2p_x + S_1S_2p_y + C_2p_z + C_2R + H$	
y_5	$S_1p_x - C_1p_y + L$	
θ_6	$atan2\left[\dfrac{o_z}{-n_z}\right]$	$\theta_2 > 0$
θ_6	$atan2\left[\dfrac{o_z}{-n_z}\right] + 180°$	$\theta_2 < 0$

NOMENCLATURE

C_i is the Cosine of the i-th joint angle

S_i is the Sine of the i-th joint angle

atan2 is the double argument arc tangent function

Table 3: Reverse Kinematic Solution of the CYRO Robot

29

$$\frac{\partial T_6}{\partial \theta_1} = \begin{bmatrix} -(C_1C_2C_6 + S_1S_6)(var1) + (S_1C_2C_6 - C_1S_6)(var2) \\ -(C_1C_2S_6 + S_1C_6)(var1) + (-S_1C_2S_6 - C_1C_6)(var2) \\ C_1S_2(var1) - S_1S_2(var2) \\ -S_2C_6 \\ S_2S_6 \\ -C_2 \end{bmatrix}$$

$$\frac{\partial T_6}{\partial \theta_2} = \begin{bmatrix} -C_2C_6(var3) + S_2C_6(var4) \\ C_2S_6(var3) - S_2C_6(var4) \\ S_2(var3) - C_2(var4) \\ -S_6 \\ -C_6 \\ 0 \end{bmatrix}$$

$$\frac{\partial T_6}{\partial x_3} = \begin{bmatrix} C_6 \\ -S_6 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \frac{\partial T_6}{\partial z_4} = \begin{bmatrix} 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\frac{\partial T_6}{\partial y_3} = \begin{bmatrix} S_6 \\ C_6 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \frac{\partial T_6}{\partial \theta_6} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

NOMENCLATURE

$$var1 = S_1(z_4S_2 + x_3C_2 - HS_2) + C_1(-y_3 + L)$$

$$var2 = C_1(z_4S_2 + x_3C_2 - HS_2) + S_1(-y_3 + L)$$

$$var3 = -z_4C_2 + x_3S_2 + HC_2$$

$$var3 = z_4S_2 + x_3C_2 - HS_2$$

Table 4: Column Vectors of the Jacobian matrix

DIFFERENTIAL JOINT COORDINATES

$$d\theta_1 = \frac{(-S_1 da_x + C_1 da_y)}{(C_1 a_x + S_1 a_y)}$$

$$d\theta_2 = \frac{NC_2 d(NS_2) - NS_2 d(NC_2)}{(NS_2)^2 + (NC_2)^2}$$

$$dz_4 = C_2 A_1 d\theta_2 + S_2 A_2 + C_2 dp_z - S_2 A_3 d\theta_2$$

$$dy_5 = C_1 p_x d\theta_1 + S_1 dp_x + S_1 p_y d\theta_1 - C_1 dp_y$$

$$d\theta_6 \approx \frac{C_6 do_z + S_6 dn_z}{S_6 o_z - C_6 n_z}$$

NOMENCLATURE

$$NC_2 = -a_z$$

$$d(NC_2) = -da_z$$

$$NS_2 = -C_1 a_x - S_1 a_y$$

$$d(NS_2) = S_1 a_x d\theta_1 - C_1 da_x - C_1 a_y d\theta_1 - S_1 da_y$$

$$A_1 = C_1 p_x + S_1 p_y$$

$$A_2 = -S_1 p_x d\theta_1 + C_1 dp_x + C_1 p_y d\theta_1 + S_1 dp_y$$

Table 5: Inverse Jacobian of the CYRO Robot

STEP	✳	+	÷	√	SIN/COS	TIME in millisec
Find the mid-seam						
Butt Joint (1)		3	3			0.384
Lap/Fillet Joint (2)	3	6				0.462
Compute segment travel time T						
Butt Joint	23	39	1	5		3.611
Lap/Fillet Joint	32	61	1	6		5.256
Find transition point						
Butt Joint	$23*N_1$	$39*N_1$	N_1	$5*N_1$		7.222
Lap/Fillet Joint	$32*N_1$	$61*N_1$	N_1	$6*N_1$		10.512
Computation of						
a Vector	9	14	5	1		1.611
o Vector (6) - (7)	15	11	3	1		1.675
n Vector (8)	3	6				0.462
Compute dT_6 Matrix		12				0.576
Normalization	9	4	3	3		1.125
Iterative algorithm (33) - (37)	$66*N_2$	$54*N_2$			$6*N_2$	7.206
Total						
Butt Joint	$62+23*N_1$ $+66*N_2$	$86+39*N_1$ $+54*N_2$	$15+N_1$	$10+N_1$	$6*N_2$	23.872
Lap/Fillet Joint	$73+23*N_1$ $+66*N_2$	$108+39*N_1$ $+54*N_2$	$15+N_1$	$10+N_1$	$6*N_2$	28.885

Table 6: Computational Requirements of the Seam Tracing Algorithm

Function	Time(μs)
Multiply	27
Add	17
Divide	39
Square Root	36
Tangent	90
Exponentiation	100
Load	10
Store	21

Table 7: Execution Times of the 8086/8087 Microprocessor(5 MHz Clock) [Intel 83]

ATE
LME